Express JS

Aruna S
Department of
Computer Science and Engineering

EXPRESS JS

Express JS Introduction

S. Aruna
Department of Computer Science and Engineering

EXPRESS JS Introduction

- ExpressJS is a web application framework that provides you with a simple API to build websites, web apps and back ends. With ExpressJS, we need not worry about low level protocols, processes, etc.
- Express provides a minimal interface to build our applications.
- It provides us the tools that are required to build our app.
- It is flexible as there are numerous modules available on **npm**, which can be directly plugged into Express.
- Express was developed by **TJ Holowaychuk** and is maintained by the Node.js foundation and numerous open source contributors.

EXPRESS JS

EXPRESS JS Features

Few of the most important features of Express.js:

- Express quickens the development pace of a web application.
- It also helps in creating mobile and web application of single-page, multi-page, and hybrid types
- Express can work with various templating engines such as Pug, Mustache, and EJS.
- Express follows the Model-View-Controller (MVC) architecture.
- It makes the integration process with databases such as MONGODB, Redis, MYSQL effortless.
- Express also defines an error-handling middleware.
- It helps in simplifying the configuration and customization steps for the application.

EXPRESS JS Installation

To install Express.js, first, you need to create a project directory and create a package.json file which will be holding the project dependencies.
Below is the code to perform the same:

npm init

To install it globally, you can use the below command:

npm install -g express

```
C:\Users\DELL>npm install express
```

To install it locally into your project folder, you need to execute the below command:

npm install express --save

HTTP Common Methods

The most common HTTP Methods are

- **GET**

The GET method requests a representation of the specified resource. Requests using GET should only retrieve data and should have no other effect.

- **POST**

The POST method creates a new object/entity of the resource identified by the URI.

- **PUT**

The PUT method modifies the existing object identified by the URI. If it does not exist then the PUT method should create one.

- **DELETE**

The DELETE method requests that the server delete the specified resource.

THANK YOU

Aruna S
Department of
Computer Science and Engineering
arunas@pes.edu